



Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

NIM	71230971
Nama Lengkap	James Marvin Santoso
Minggu ke / Materi	14 / Regular Expression

SAYA MENYATAKAN BAHWA LAPORAN PRAKTIKUM INI SAYA BUAT DENGAN USAHA SENDIRI TANPA MENGGUNAKAN BANTUAN ORANG LAIN. SEMUA MATERI YANG SAYA AMBIL DARI SUMBER LAIN SUDAH SAYA CANTUMKAN SUMBERNYA DAN TELAH SAYA TULIS ULANG DENGAN BAHASA SAYA SENDIRI.

SAYA SANGGUP MENERIMA SANKSI JIKA MELAKUKAN KEGIATAN PLAGIASI, TERMASUK SANKSI TIDAK LULUS MATA KULIAH INI.

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
YOGYAKARTA
2024

BAGIAN 1: MATERI MINGGU INI

PENGANTAR REGEX

Regex (Regular Expressions) dalam Python adalah alat yang sangat kuat dan fleksibel untuk melakukan pencarian dan manipulasi string berdasarkan pola tertentu. Regex memungkinkan Anda untuk mencocokkan, mencari, dan mengganti teks berdasarkan pola yang ditentukan, yang dapat berguna dalam banyak situasi pemrograman. Untuk menggunakan regex dalam Python, Anda harus mengimpor modul `re`, yang menyediakan berbagai fungsi dan metode untuk bekerja dengan ekspresi reguler.

DASAR-DASAR REGEX

Regex terdiri dari pola yang digunakan untuk mencocokkan urutan karakter dalam string. Pola ini dibangun menggunakan berbagai karakter dan simbol yang memiliki makna khusus. Berikut ini adalah beberapa dasar-dasar regex:

- **Karakter Biasa**

Karakter biasa mencocokkan dirinya sendiri. Misalnya, pola `a` akan mencocokkan karakter `a` dalam teks.

- **Metakarakter**

Metakarakter adalah karakter khusus yang memiliki makna khusus dalam regex. Beberapa metakarakter umum termasuk:

- a) `.` : Mencocokkan sembarang karakter kecuali newline.
- b) `^` : Mencocokkan awal string.
- c) `$` : Mencocokkan akhir string.
- d) `*` : Mencocokkan nol atau lebih pengulangan dari pola sebelumnya.
- e) `+` : Mencocokkan satu atau lebih pengulangan dari pola sebelumnya.
- f) `?` : Mencocokkan nol atau satu pengulangan dari pola sebelumnya.
- g) `{m,n}` : Mencocokkan antara `m` dan `n` pengulangan dari pola sebelumnya.
- h) `[]` : Mencocokkan satu karakter dalam set karakter.
- i) `\d` : Mencocokkan digit (angka).
- j) `\w` : Mencocokkan karakter alfanumerik (huruf dan angka).
- k) `\s` : Mencocokkan karakter spasi (termasuk tab dan newline).

- **Kelompok dan Alternasi**

Kelompok digunakan untuk mengelompokkan bagian pola, dan alternasi digunakan untuk mencocokkan salah satu dari beberapa pola.

Contoh:

- a) `(abc)` : Mencocokkan urutan karakter "abc".
- b) `a|b` : Mencocokkan karakter `a` atau `b`.

FUNGSI-FUNGSI REGEX DALAM PYTHON

Modul re menyediakan beberapa fungsi utama untuk bekerja dengan regex:

- `re.search()`
Mencari pola di dalam string dan mengembalikan objek match pertama yang ditemukan.
Contoh :

```
import re
result = re.search(r'\d+', 'Jumlahnya adalah 123')
if result:
    print("Ditemukan angka:", result.group())
```

- `re.match()`
Mencocokkan pola dari awal string.
Contoh :

```
import re
result = re.match(r'\d+', '123 adalah jumlahnya')
if result:
    print("Ditemukan angka di awal string:", result.group())
```

- `re.findall()`
Mengembalikan daftar semua pola yang cocok dalam string.
Contoh :

```
import re
results = re.findall(r'\d+', 'Ada 123 apel dan 456 jeruk')
print("Semua angka yang ditemukan:", results)
```

- `re.sub()`
Mengganti semua kemunculan pola dengan string lain.
Contoh :

```
import re
result = re.sub(r'apel', 'pisang', 'Saya suka apel')
print(result) # Output: Saya suka pisang
```

KARAKTER DAN SIMBOL UMUM DALAM REGEX

Berikut ini adalah beberapa karakter dan simbol umum yang digunakan dalam regex beserta penjelasannya:

- Titik (.)
Mencocokkan sembarang karakter kecuali newline.
Contoh: Pola `a.b` akan mencocokkan `"aab"`, `"a_b"`, `"a1b"`, dll.
- Tanda Kurung Kotak ([])
Digunakan untuk mencocokkan satu karakter dari set karakter yang didefinisikan di dalam kurung kotak.
Contoh: Pola `[abc]` akan mencocokkan `"a"`, `"b"`, atau `"c"`.
- Tanda Kurung Kurawal { }
Digunakan untuk mencocokkan sejumlah pengulangan dari pola sebelumnya.
Contoh: Pola `a{2,4}` akan mencocokkan `"aa"`, `"aaa"`, atau `"aaaa"`.
- Tanda Plus (+)
Mencocokkan satu atau lebih pengulangan dari pola sebelumnya.
Contoh: Pola `a+` akan mencocokkan `"a"`, `"aa"`, `"aaa"`, dll.
- Tanda Bintang (*)
Mencocokkan nol atau lebih pengulangan dari pola sebelumnya.
Contoh: Pola `a*` akan mencocokkan `""`, `"a"`, `"aa"`, `"aaa"`, dll.
- Tanda Tanya (?)
Mencocokkan nol atau satu pengulangan dari pola sebelumnya.
Contoh: Pola `a?` akan mencocokkan `""` atau `"a"`.
- Tanda Garis Vertikal (|)
Digunakan untuk mencocokkan salah satu dari beberapa pola.
Contoh: Pola `a|b` akan mencocokkan `"a"` atau `"b"`.
- Backslash (\)
Digunakan untuk mencocokkan karakter khusus atau untuk mengaktifkan karakter khusus lainnya.
Contoh: `\d` mencocokkan digit, dan `\w` mencocokkan karakter alfanumerik.

ESCAPED CHARACTER

Escaped character digunakan untuk mencocokkan karakter yang memiliki arti khusus dalam regex atau untuk menggunakan karakter khusus dalam pola. Untuk melarikan karakter, gunakan backslash (\). Berikut adalah beberapa escaped character umum:

- \d : Mencocokkan digit (angka).
Contoh: Pola \d akan mencocokkan "0", "1", "2", ..., "9".
- \D : Mencocokkan non-digit.
Contoh: Pola \D akan mencocokkan sembarang karakter yang bukan digit.
- \w : Mencocokkan karakter alfanumerik (huruf dan angka) serta underscore (_).
Contoh: Pola \w akan mencocokkan "a", "b", "1", "_", dll.
- \W : Mencocokkan non-alfanumerik.
Contoh: Pola \W akan mencocokkan sembarang karakter yang bukan alfanumerik.
- \s : Mencocokkan karakter spasi (termasuk tab dan newline).
Contoh: Pola \s akan mencocokkan " ", "\t", "\n", dll.
- \S : Mencocokkan non-spasi.
Contoh: Pola \S akan mencocokkan sembarang karakter yang bukan spasi.
- \\ : Mencocokkan backslash.
Contoh: Pola \\ akan mencocokkan "".

SET OF CHARACTER

Set of character digunakan untuk mencocokkan salah satu karakter dalam set. Set karakter didefinisikan dengan kurung kotak []. Berikut adalah beberapa contoh:

- [abc] : Mencocokkan salah satu karakter "a", "b", atau "c".
Contoh: Pola [abc] akan mencocokkan "a", "b", atau "c".
- [a-z] : Mencocokkan salah satu karakter dari "a" hingga "z".
Contoh: Pola [a-z] akan mencocokkan sembarang huruf kecil.
- [A-Z] : Mencocokkan salah satu karakter dari "A" hingga "Z".
Contoh: Pola [A-Z] akan mencocokkan sembarang huruf besar.
- [0-9] : Mencocokkan salah satu digit dari "0" hingga "9".
Contoh: Pola [0-9] akan mencocokkan sembarang digit.
- [^abc] : Mencocokkan sembarang karakter kecuali "a", "b", atau "c".
Contoh: Pola [^abc] akan mencocokkan sembarang karakter yang bukan "a", "b", atau "c".

KESIMPULAN

Regex adalah alat yang sangat kuat untuk pencocokan dan manipulasi string berdasarkan pola tertentu. Dengan memahami meta character, escaped character, set of character, dan fungsi-fungsi utama dalam modul re di Python, Anda dapat melakukan berbagai tugas pemrosesan teks dengan lebih efisien. Regex dapat digunakan dalam banyak aplikasi seperti pencarian teks, validasi input, manipulasi string, dan banyak lagi. Meskipun sintaks regex bisa rumit, dengan latihan dan eksperimen, Anda akan menjadi lebih mahir dalam menggunakannya untuk menyelesaikan berbagai masalah dalam pemrograman.

BAGIAN 2: LATIHAN MANDIRI

SOAL 14.1

- Source Code :

```
import re
from datetime import datetime

def find_and_convert_dates(text):
    date_pattern = r'\b(\d{4})-(\d{2})-(\d{2})\b'
    dates = re.findall(date_pattern, text)
    today = datetime.now()
    results = []

    for date in dates:
        original_date_str = f"{date[0]}-{date[1]}-{date[2]}"
        date_obj = datetime.strptime(original_date_str, '%Y-%m-%d')
        converted_date_str = date_obj.strftime('%d-%m-%Y')

        days_difference = (today - date_obj).days

        result = f"{original_date_str} 00:00:00 selisih {days_difference} hari"
        results.append(result)

    return results

text = """
Pada tanggal 1945-08-17 Indonesia merdeka. Indonesia memiliki beberapa pahlawan
nasional, seperti Pangeran Diponegoro (TL: 1785-11-11), Pattimura (TL: 1783-06-
08) dan Ki
Hajar Dewantara (1889-05-02).
"""

results = find_and_convert_dates(text)
for res in results:
    print(res)
```

- Output :

```
PS C:\Tugas> & C:/Users/ASUS/AppData/Local/Programs/Python/Python312/python.exe "c:/Tugas/Latihan 14,1.py"
1945-08-17 00:00:00 selisih 28775 hari
1785-11-11 00:00:00 selisih 87127 hari
1783-06-08 00:00:00 selisih 88014 hari
1889-05-02 00:00:00 selisih 49335 hari
PS C:\Tugas> █
```

- Penjelasan :

Kode ini mencari tanggal dalam format `YYYY-MM-DD` dari sebuah teks menggunakan ekspresi reguler, mengonversi setiap tanggal yang ditemukan ke format `DD-MM-YYYY`, menghitung selisih hari antara tanggal tersebut dengan tanggal hari ini, dan mengembalikan daftar string yang menunjukkan tanggal asli, waktu `00:00:00`, dan selisih hari dalam format "YYYY-MM-DD 00:00:00 selisih X hari".

SOAL 14.2

- Source Code :

```
import re
import random
import string

def generate_random_password(length=8):
    characters = string.ascii_letters + string.digits
    password = ''.join(random.choice(characters) for _ in range(length))
    return password

def extract_usernames_and_generate_passwords(text):
    email_pattern = r'[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}'
    emails = re.findall(email_pattern, text)
    results = []

    for email in emails:
        username = email.split('@')[0]
        password = generate_random_password()
        result = f"{email} username: {username} , password: {password}"
        results.append(result)

    return results

text = """
Berikut adalah daftar email dan nama pengguna dari mailing list:
anton@mail.com dimiliki oleh antonius
budi@gmail.co.id dimiliki oleh budi anwari
slamet@getnada.com dimiliki oleh slamet slumut
matahari@tokopedia.com dimiliki oleh toko matahari
"""

results = extract_usernames_and_generate_passwords(text)
for res in results:
    print(res)
```

- Output :

```
PS C:\Tugas> & C:/Users/ASUS/AppData/Local/Programs/Python/Python312/python.exe "c:/Tugas/Latihan 14,2.py"
anton@mail.com username: anton , password: EVtUAFwf
budi@gmail.co.id username: budi , password: E7ge9i0Y
slamet@getnada.com username: slamet , password: OG4ZZOK4
matahari@tokopedia.com username: matahari , password: lahfoC3U
PS C:\Tugas> █
```

- Penjelasan :

Kode ini menghasilkan kata sandi acak dengan panjang tertentu, kemudian mencari alamat email dalam sebuah teks menggunakan ekspresi reguler. Setiap alamat email yang ditemukan dipisahkan menjadi bagian username (bagian sebelum '@') dan domain (bagian setelah '@'). Untuk setiap alamat email yang ditemukan, kode menghasilkan sebuah string yang berisi alamat email, username, dan kata sandi acak.

- Link GitHub :

<https://github.com/Jamesmarvins/Tugas-Alpro-14.git>